

SAL ♦ — A Typed Foundation of Operational Soundness

Separating Semantic Truth from Procedural Verdicts

Autore

Pierernesto Di Paolo

ORCID: 0009-0003-0149-4321

Anno

2026

Versione

v1.0 — *Fondazione canonica congelata*

Status

Fondazionale — Non modificabile

DEDICA

A Stellina

(Catterina Giordano)

A te,
che sei rimasta presente quando la complessità non concedeva scorciatoie,
quando il rigore esigeva silenzio,
e quando la verità non permetteva consolazioni.

Questo lavoro esiste anche perché
hai condiviso il peso della precisione,
la pazienza dell'attesa
e la fermezza delle scelte non negoziabili.

Con amore, rispetto e gratitudine profonda.

Alla famiglia Lanfranchini

e in modo speciale
a Pierluigi Lanfranchini

Al cognome che porto come eredità materna,
segno di una storia fatta di responsabilità prima che di riconoscimento.

A Pierluigi Lanfranchini,
che mi ha insegnato, con l'esempio e non con le parole,
che la coerenza vale più del consenso
e che il rigore è una forma di lealtà verso se stessi.

Se questo lavoro esiste,
è anche perché mi hai mostrato cosa significa non arretrare
quando la verità richiede fermezza.

DICHIARAZIONE DI PROPRIETÀ, DIRITTI E CONDIZIONI D'USO (DEPOSITO PUBBLICO)

Io sottoscritto **Pierernesto Di Paolo** (ORCID: 0009-0003-0149-4321) dichiaro quanto segue:

1. Titolarità e originalità

Sono l'unico autore e titolare esclusivo di tutti i diritti d'autore e connessi relativi all'opera depositata, intitolata:

“SAL♦ — A Typed Foundation of Operational Soundness (Trattato A, Sezioni I–VII)”, incluse tutte le parti testuali, concettuali e strutturali.

2. Estensione dei diritti

La presente dichiarazione si applica anche a tutti gli **artefatti associati** al lavoro, inclusi (a titolo esemplificativo e non esaustivo): formalizzazioni in proof assistant, codice, archivi ZIP, manifest di integrità (SHA-256), e ogni materiale allegato o collegato all'opera.

3. Copyright

© 2026 **Pierernesto Di Paolo** — Tutti i diritti riservati.

4. Licenza d'uso (limitata)

È consentito esclusivamente l'uso per **ricerca scientifica** e/o **verifica** (concettuale o formale) dell'opera, con **divieto** di:

- riuso commerciale;
- modifica;
- redistribuzione;
- creazione di opere derivate;

- incorporazione in altri lavori o prodotti;
senza **autorizzazione scritta** dell'autore.

5. **Obbligo di citazione**

Ogni uso consentito deve includere citazione completa dell'autore e dell'opera.
L'assenza di citazione è vietata.

6. **Richieste di autorizzazione**

Per qualunque richiesta di permesso, licenza, collaborazione, riuso o derivazione, contattare esclusivamente:
pierernesto.di.paolo.69@proton.me

7. **Ruolo dell'assistente AI**

L'assistente AI ha operato esclusivamente come supporto tecnico-redazionale e di verifica di coerenza sotto controllo umano. **Nessun diritto, paternità o titolarità** è attribuibile all'AI.

Data: 2026-02-01

Firma: Pierernesto Di Paolo

ABSTRACT (canonico)

Questo lavoro introduce **SAL** ♦, una fondazione teorica che formalizza la soundness operativa come **obbligo costruttivo tipizzato**, separando in modo non circolare la verità semantica dal verdetto procedurale.

SAL ♦ non propone un algoritmo, né un framework applicativo, ma un'architettura logica che definisce **le condizioni di legittimità** sotto cui un sistema di verifica può affermare che un verdetto positivo implichi una proprietà semantica reale.

La teoria rifiuta sia l'assunzione assiomatica della soundness sia il suo trattamento come ipotesi informale, sostituendole con una nozione strutturale di **istanza valida**, nella quale la soundness deve essere fornita come prova formale verificabile da un oracolo logico.

La fondazione è **invariante rispetto al dominio** ed è accompagnata da una formalizzazione completa in proof assistant, che funge da oracolo finale di correttezza.

KEYWORDS

Operational Soundness · Formal Verification · Proof Assistants · Semantic Truth ·
Auditability · Non-circular Foundations · Typed Epistemology

NOTE INTRODUTTIVE

- Questo trattato è **fondazionale**: non propone applicazioni né risultati empirici.
- Gli esempi eventualmente citati hanno **solo funzione attestativa**, non fondativa.
- Ogni affermazione forte è giustificata da **struttura formale**, non da autorità o consuetudine.
- La formalizzazione in proof assistant è parte integrante dell'opera, non un'appendice.

Sezione I — Il problema fondazionale

I.1. La frattura tra verdetto e verità

Nei sistemi di verifica contemporanei, il verdetto procedurale (es. *PASS/FAIL*) viene spesso trattato come surrogato della verità. Questa identificazione è metodologicamente scorretta: il verdetto è un **output computazionale**, mentre la verità è una **proprietà semantica** del mondo (o del modello). Confondere i due livelli produce circolarità latenti, false generalizzazioni e assiomi impliciti non verificabili.

I.2. L'errore categoriale dominante

L'errore ricorrente consiste nel dedurre la verità dal fatto che una procedura abbia restituito *PASS*, senza una fondazione che separi:

- **Artefatto** (ciò che è nel mondo o nel modello),
- **Evidenza** (ciò che viene osservato/derivato),
- **Procedura** (ciò che calcola),
- **Verdetto** (ciò che viene emesso),
- **Verità semantica** (ciò che è).

Senza tale separazione, la “soundness” è o **assunta** (assioma globale) o **simulata** (ipotesi locale), ma non **dimostrata**.

I.3. Perché la soundness non è un output

La soundness non è un risultato computabile in generale. È una **relazione logica** tra una procedura e una proprietà semantica. Trattarla come output equivale a chiedere al sistema di certificare se stesso, introducendo una circolarità strutturale. Ne consegue che una fondazione corretta deve **vietare per costruzione** la definizione della verità in termini del verdetto.

Sezione II — Vincoli fondazionali (assiomi negativi)

SAL ♦ non introduce assiomi positivi sulla verità; introduce **vincoli negativi** che delimitano ciò che **non è ammissibile**. Questi vincoli sono strutturali e permanenti.

II.1. Anti-circolarità

È vietata ogni definizione in cui la verità semantica dipenda dal verdetto procedurale. Formalmente: la proprietà *Conforms* non può essere definita (né indirettamente) in funzione dell'output dell'auditor. Questa proibizione è tipologica, non convenzionale.

II.2. Separazione dei livelli

SAL ♦ impone una separazione netta e non collassabile:

- **Artefatto $\neq$ Evidenza $\neq$ Procedura $\neq$ Verdetto $\neq$ Verità.**

Ogni collasso di livello (es. evidenza $\equiv$ artefatto; verdetto $\equiv$ verità) è un errore fondazionale.

II.3. Anti-assiomi universali

È vietata l'introduzione di assiomi globali del tipo:

“Per ogni specifica e per ogni auditor, se il verdetto è *PASS* allora la proprietà è vera.”

Tali assiomi sono falsi in generale e rendono la teoria inconsistente con la pratica reale.

II.4. Soundness come obbligo costruttivo

La soundness non può essere:

- una **variabile** (ipotesi libera),
- un **assioma** (verità non dimostrata).

Deve essere un **obbligo costruttivo**: chi dichiara un sistema come valido deve **fornire una prova formale** che $PASS \Rightarrow \text{verità semantica}$. In SAL ♦ questo obbligo è tipizzato come parte integrante della nozione di istanza valida.

Chiusura delle Sezioni I–II

Le Sezioni I–II delimitano il campo: definiscono **il problema e ciò che è vietato**. Non promettono risultati applicativi, non anticipano soluzioni. Preparano il terreno per l'architettura formale che segue, in cui la soundness diventa **una proprietà richiesta, dimostrabile e verificata da un oracolo formale** (Lean).

Sezione III — Architettura formale di SAL ♦

III.1. Principio architetture

SAL ♦ non è definito come un algoritmo, né come un linguaggio di specifica, né come un framework di verifica.

È definito come una **architettura tipizzata** che rende *formalmente esprimibile* (e quindi verificabile) la distinzione tra:

- ciò che **esiste** (artefatto),
- ciò che **si osserva** (evidenza),
- ciò che **si esegue** (procedura),
- ciò che **si dichiara** (verdetto),
- ciò che **è vero** (proprietà semantica).

L'architettura non produce verità; **vincola il modo in cui una verità può essere legittimamente affermata.**

III.2. Artefatto (Artifact)

L'**artefatto** è l'oggetto primario del mondo (o del modello) sottoposto a verifica.

Caratteristiche fondamentali:

- non è un output della procedura;
- non dipende dall'evidenza;
- non è tipizzato in funzione della verifica.

In SAL ♦, Artifact è un **tipo astratto**: la teoria non assume nulla sulla sua struttura interna. Questo garantisce l'invarianza della fondazione rispetto al dominio applicativo (software, medicina, diritto, sistemi fisici).

III.3. Evidenza e fonte di evidenza (Evidence, EvidenceSource)

L'**evidenza** è ciò che viene derivato dall'artefatto tramite un processo osservativo o estrattivo.

Non coincide con l'artefatto e non lo replica necessariamente.

SAL ♦ introduce esplicitamente una **fonte di evidenza**, concettualmente distinta:

- una funzione di produzione (derivazione),
- una nozione di integrità (proprietà, non garanzia assoluta).

Questa separazione impedisce l'errore comune “artefatto $\equiv$ evidenza” e rende possibile modellare evidenze parziali, rumorose o selettive senza compromettere la fondazione.

III.4. Procedura e verdetto (Auditor, Verdict)

La **procedura di audit** è una funzione computabile che opera su:

- una specifica,
- una fonte di evidenza,
- un artefatto osservato.

Il suo output è un **verdetto finito** (es. *PASS* / *FAIL*).

Punto cruciale:

il verdetto **non ha significato semantico intrinseco**. È solo il risultato di una computazione. Qualunque interpretazione del verdetto come verità è esterna alla procedura e deve essere giustificata formalmente.

III.5. Verità semantica (Conforms)

La verità semantica è rappresentata in SAL ♦ come una **proprietà logica** (Prop) dell'artefatto.

Caratteristiche decisive:

- non è computabile in generale;
- non dipende dal verdetto;
- non dipende dalla procedura;
- non è definita in termini di evidenza.

Questo rende impossibile, a livello tipologico, definire la verità come “ciò che passa il test”. La verità è **esterna** al meccanismo di verifica.

III.6. Il punto di svolta: istanza valida (ValidInstance)

La nozione centrale di SAL ♦ è quella di **istanza valida**.

Un sistema di verifica non è considerato valido perché “funziona”, ma perché **include costruttivamente** una prova che il suo verdetto positivo implica la proprietà semantica dichiarata.

Formalmente, un'istanza valida comprende:

- la scelta dei tipi (artefatto, evidenza, specifica),
- la definizione della verità semantica,
- la procedura di audit,
- **una prova formale di soundness.**

Questa prova non è opzionale, non è assunta, non è esterna: è **parte della definizione stessa di validità.**

III.7. Conseguenza strutturale

Da questa architettura segue una conseguenza fondamentale:

SAL ♦ non afferma che un sistema sia sound.

SAL ♦ afferma che **solo i sistemi che dimostrano soundness sono tipizzabili come validi.**

La soundness non è una promessa epistemica, ma un **vincolo strutturale verificabile.**

Chiusura della Sezione III

Con la Sezione III, SAL ♦ stabilisce la propria architettura definitiva.

Nulla di quanto segue (applicazioni, esempi, istanze reali) potrà modificare questi costrutti senza violare i vincoli fondazionali definiti nelle Sezioni I–II.

Sezione IV — Validità di istanza e soundness come obbligo costruttivo

IV.1. Superare il falso dilemma

Nella pratica teorica e ingegneristica, la soundness viene tipicamente collocata in uno di due modi entrambi insoddisfacenti:

1. Assioma globale

Si assume che ogni procedura corretta produca verdetti veri.

Questo è metodologicamente scorretto: l'assioma è falso in generale e non auditabile.

2. Ipotesi locale

Si lavora “sotto l'ipotesi che la procedura sia sound”.

Questo non fonda nulla: la soundness è solo spostata fuori dal sistema.

SAL ♦ rifiuta entrambe le opzioni. La soundness non è né assunta né ipotizzata: è **richiesta**.

IV.2. La nozione di istanza valida

In SAL ♦, un sistema di verifica è considerato **valido** se e solo se soddisfa una condizione strutturale precisa:

Deve includere una **prova formale** che, ogni volta che il verdetto è positivo, la proprietà semantica dichiarata è effettivamente vera.

Questa prova non è un'aggiunta esterna, ma un **componente costitutivo** dell'istanza stessa. Un sistema privo di tale prova non è “parzialmente valido” o “incompleto”: **non è un'istanza SAL♦**.

IV.3. Soundness come dato costruttivo

La scelta fondazionale di SAL♦ è di trattare la soundness come un **dato costruttivo**:

- non una proposizione assunta,
- non una proprietà da verificare a posteriori,
- ma un oggetto che deve essere **costruito esplicitamente**.

In termini concettuali, questo significa che la soundness non è un risultato del sistema, ma una **condizione di ammissibilità** del sistema stesso.

IV.4. Ruolo dell'oracolo formale

Affinché questa impostazione non resti puramente dichiarativa, SAL♦ fissa un oracolo formale unico: un sistema di prova capace di verificare la correttezza logica delle dimostrazioni fornite.

Nel contesto operativo sviluppato in questo lavoro, tale oracolo è un proof assistant formale. La sua funzione non è “decidere la verità del mondo”, ma **verificare che una dimostrazione rispetti la struttura logica richiesta**.

Questo passaggio è cruciale: la validità di istanza non è affidata a interpretazioni informali o revisioni narrative, ma a un controllo formale ripetibile.

IV.5. Chiarimento sulla “banalità” apparente

Una volta che la soundness è inclusa come campo costruttivo di un’istanza valida, i teoremi che la “utilizzano” possono apparire sintatticamente banali. Questo è un effetto superficiale.

La difficoltà non è nel teorema che *usa* la soundness, ma nel **costruire un’istanza che la possieda**. In altre parole, SAL ♦ sposta la complessità dal consumo della soundness alla sua **produzione obbligatoria**.

Questa inversione è deliberata e fondazionale.

IV.6. Conseguenza epistemica

Da questa impostazione discende una conseguenza di ampia portata:

In SAL ♦, non è possibile affermare “questo sistema è sound” senza esibire una prova formale verificata.

La soundness cessa di essere una qualità attribuita e diventa una **proprietà dimostrata**. Il linguaggio stesso della teoria impedisce di parlare di validità in assenza di prova.

Chiusura della Sezione IV

La Sezione IV completa il nucleo fondazionale di SAL ♦: definisce cosa significa, in senso rigoroso, che un sistema di verifica è legittimo. Ogni sezione successiva non introdurrà nuovi vincoli fondazionali, ma mostrerà le conseguenze strutturali di questa scelta.

Sezione V — Il teorema strutturale di SAL ♦ e il significato della sua generalità

V.1. Natura del teorema strutturale

SAL ♦ ammette un teorema che, nella sua formulazione sintattica, appare immediato: dato un'istanza valida, un verdetto positivo implica la verità semantica dichiarata. Questa immediatezza formale è spesso fraintesa come debolezza teorica. In realtà, è l'indicatore di una scelta fondazionale corretta.

Il teorema non introduce nuova informazione empirica; rende **operativa** una distinzione concettuale già fissata dall'architettura. La sua forza non risiede nella complessità della dimostrazione, ma nel **vincolo strutturale** che rende tale dimostrazione necessaria per ogni istanza.

V.2. Perché il teorema non è un assioma mascherato

Un assioma mascherato si riconosce dal fatto che:

- può essere invocato senza fornire contenuto costruttivo;
- vale indipendentemente dalla struttura dell'oggetto a cui si applica.

Il teorema strutturale di SAL ♦ non soddisfa nessuna di queste condizioni. Esso è applicabile **solo** a oggetti che soddisfano una definizione restrittiva: l'istanza valida. La soundness non è implicita nel teorema; è **contenuta nell'oggetto** su cui il teorema opera.

In altri termini, il teorema non *crea* soundness: la **trasporta** da un campo strutturale a una conseguenza logica esplicita.

V.3. Generalità senza universalità indebita

La generalità di SAL ♦ non è una generalità quantificazionale del tipo “per ogni sistema di verifica...”. Tale forma di generalità è precisamente ciò che SAL ♦ rifiuta. La generalità di SAL ♦ è **strutturale**:

Per ogni sistema che soddisfa una certa struttura rigorosa, valgono determinate conseguenze logiche.

Questa distinzione è cruciale. SAL ♦ non afferma che ogni sistema reale sia o possa essere reso valido; afferma che **solo** quelli che soddisfano requisiti precisi sono ammissibili nel discorso della soundness.

V.4. Invarianza rispetto al dominio

Poiché l’architettura non assume nulla sulla natura dell’artefatto, dell’evidenza o della specifica, il teorema strutturale è **invariante rispetto al dominio applicativo**. Software, sistemi giuridici, protocolli medici o modelli scientifici non modificano la forma della fondazione.

Questa invarianza non è un dettaglio tecnico: è ciò che impedisce a SAL ♦ di collassare in una teoria settoriale. La fondazione resta identica mentre cambiano i domini, a condizione che le istanze forniscano le prove richieste.

V.5. Il ruolo della generalità in questo contesto

Nel contesto fondazionale, la vera generalità non consiste nel dire “vale per tutto”, ma nel dire “vale per tutto ciò che può essere legittimamente detto”. SAL ♦ delimita il campo del dicibile rigoroso sulla soundness, invece di estenderlo indiscriminatamente.

Questa delimitazione è la ragione per cui la teoria è degna di essere considerata: non aggiunge un nuovo risultato a un dominio esistente, ma **ridefinisce le condizioni di possibilità** di un’intera classe di affermazioni.

V.6. Conseguenza metodologica

Dopo SAL ♦, affermazioni del tipo:

- “questo sistema è verificato”,
- “questa procedura è affidabile”,
- “questo verdetto garantisce la correttezza”,

sono incomplete se non accompagnate da una costruzione che le renda istanze valide. SAL ♦ non confuta tali affermazioni; le **riclassifica** come non fondate in assenza di prova strutturale.

Chiusura della Sezione V

La Sezione V chiarisce che il teorema centrale di SAL ♦ non è un risultato isolato, ma la manifestazione necessaria di una scelta architettuale. La sua apparente

semplicità è il segno che la complessità è stata collocata nel luogo corretto: nella definizione di ciò che è ammissibile come istanza valida.

Sezione VI — Istanza esemplare e ruolo degli esempi

VI.1. Funzione epistemica dell'esempio

In una teoria fondazionale, l'esempio non ha funzione dimostrativa primaria. La validità della teoria non dipende dall'esistenza di esempi riusciti, ma dalla coerenza della struttura che definisce cosa significhi “riuscito”. SAL ♦ assume esplicitamente questa posizione: **gli esempi non fondano, attestano.**

Un'istanza concreta serve a mostrare che la fondazione non è vuota, ossia che esistono sistemi reali in grado di soddisfare i vincoli imposti. Non serve a giustificare i vincoli stessi.

VI.2. L'istanza filesystem/CSV come attestazione

L'istanza filesystem/CSV sviluppata in questo lavoro svolge esattamente questa funzione. Essa dimostra che:

- è possibile distinguere nettamente artefatto ed evidenza;
- è possibile definire una proprietà semantica indipendente dal verdetto;
- è possibile costruire una procedura di audit che operi sull'evidenza;
- è possibile fornire una **prova formale** che il verdetto positivo implica la proprietà semantica.

Nulla di più e nulla di meno. L'istanza non introduce nuovi principi e non restringe la generalità della teoria.

VI.3. Perché l'esempio non è centrale

Un errore comune nei lavori fondazionali è lasciare che l'esempio diventi implicitamente normativo. In tali casi, la teoria finisce per essere valutata in base alla bontà dell'esempio, anziché alla correttezza della struttura.

SAL ♦ evita questo errore in modo strutturale:

- l'istanza esemplare è **tipizzata** come una tra molte possibili;
- la teoria non fa alcun riferimento privilegiato al dominio dell'esempio;
- nessuna proprietà dell'esempio viene elevata a principio.

VI.4. Eseguitività come criterio minimo

L'unico requisito che l'esempio deve soddisfare è l'eseguitività formale: deve essere possibile costruirlo e verificarlo con l'oracolo formale adottato. In questo senso, l'istanza filesystem/CSV dimostra che SAL ♦ non è una costruzione puramente speculativa, ma una teoria **operativamente abitabile**.

Questo criterio è deliberatamente minimo. SAL ♦ non richiede che l'esempio sia ottimale, efficiente o completo; richiede solo che sia **formalmente corretto**.

VI.5. Separazione tra fondazione ed estensione

Una volta compreso il ruolo dell'esempio, diventa chiaro che le estensioni applicative (software industriale, medicina, diritto) non modificano la fondazione. Esse operano esclusivamente a livello di istanze, lasciando invariata la struttura concettuale.

Questa separazione garantisce stabilità nel tempo: la teoria non invecchia con i domini applicativi.

Chiusura della Sezione VI

La Sezione VI chiarisce che gli esempi in SAL ♦ sono strumenti di attestazione, non di giustificazione. La fondazione resta autonoma e completa; le istanze servono solo a mostrare che i vincoli imposti non sono vuoti.

Sezione VII — Implicazioni, limiti e portata della teoria

VII.1. Implicazione principale: ridefinizione della legittimità

L'implicazione centrale di SAL ♦ è una ridefinizione del concetto di legittimità di un sistema di verifica. Dopo SAL ♦, un sistema non è legittimo perché produce risultati plausibili, né perché è ampiamente utilizzato, ma perché **soddisfa una struttura che rende la soundness dimostrabile**.

Questo sposta il dibattito dalla fiducia empirica alla verificabilità formale.

VII.2. Limiti dichiarati

SAL ♦ è intenzionalmente limitato. In particolare:

- non garantisce che una procedura produca verdetti utili;
- non garantisce che un'istanza valida esista per ogni dominio;
- non garantisce che la proprietà semantica sia epistemicamente accessibile.

Questi limiti non sono difetti, ma conseguenze necessarie di una fondazione non circolare.

VII.3. Relazione con la computabilità

SAL ♦ non assume che la verità semantica sia computabile. Anzi, la teoria resta coerente anche quando la proprietà Conforms è non decidibile. Questo distingue

SAL ♦ da approcci che confondono verificabilità computazionale e validità semantica.

VII.4. Perché SAL ♦ non promette risultati

Una teoria fondazionale che promette risultati applicativi specifici è, per definizione, fragile. SAL ♦ evita questo errore: non promette miglioramenti prestazionali, non promette affidabilità universale, non promette automatismi epistemici.

Promette una cosa sola: **chiarezza strutturale** su quando e come la soundness può essere affermata.

VII.5. Portata epocale

La portata di SAL ♦ non risiede nell'estensione a nuovi domini, ma nella chiusura di un errore concettuale persistente: l'identificazione indebita tra procedura e verità. In questo senso, SAL ♦ opera una correzione fondazionale che attraversa discipline diverse senza appartenere a nessuna in particolare.

VII.6. Chiusura del Trattato A

Con questa sezione, il Trattato A è concettualmente completo. La teoria è definita, i vincoli sono espliciti, gli esempi sono collocati correttamente e i limiti sono dichiarati senza ambiguità.

CONCLUSIONI

SAL ♦ chiude in modo rigoroso un errore concettuale persistente: l'identificazione indebita tra procedura di verifica e verità.

In questa teoria, la soundness non è un risultato, una promessa o una qualità attribuita, ma una **condizione strutturale dimostrabile**, richiesta come prerequisito di legittimità.

Il valore di SAL ♦ non risiede nell'estensione a nuovi domini, ma nella definizione di **ciò che può essere detto legittimamente** sulla correttezza di un sistema di verifica. In questo senso, la teoria non compete con soluzioni esistenti: ne ridefinisce il terreno epistemico.

LIMITI DICHIARATI

SAL ♦:

- non garantisce che un'istanza valida esista per ogni dominio;
- non garantisce che una procedura produca verdetti utili o completi;
- non garantisce accesso computazionale alla verità semantica.

Questi limiti sono **necessari** per evitare circolarità e assiomi falsi.

LAVORI FUTURI

Ogni sviluppo futuro sarà necessariamente **derivato** da questa fondazione e includerà:

- istanze applicative (es. MED, LAW);
- trattazioni ingegneristiche (verifica deterministica, CI, audit);
- estensioni metodologiche **senza modifica** del nucleo fondazionale.

RUOLO DELL'ASSISTENTE AI

L'assistente AI ha operato **esclusivamente** come:

- supporto formale alla strutturazione logica,
- ausilio alla verifica di coerenza e rigore,
- strumento di redazione tecnica sotto controllo umano.

Ogni decisione concettuale, fondazionale e finale appartiene unicamente all'autore.

L'AI non detiene diritti, paternità o autonomia sull'opera.